# Class Diagram Justification

## Class Diagram

Since submitting our original class diagram, the structure of the project has remained largely the same.

The scope of the Model class has largely outgrown its purpose, and as such we intend to split up the functionality into discrete classes, such as the timeUntilCollision method.

Since the project structure has not changed since the start of development, it is likely to change before final submission, with issues such as coupling left to be explored and corrected.

The build menu and run menu classes have not been fleshed out, as they are placeholders for the related fxml files for the Build and Run menu controllers.

## Explanation of Additions

MouseEvent handlers – All gizmos have a corresponding mouse event handler, to allow for connection of triggers and actions between them, as well as placing them on the board during build mode.

In addition to this, features such as rotating, (dis)connecting, moving, and deleting have MouseEvent handlers to facilitate such functionality in the build view.

Type enum - Gizmos have been given an enum type to allow for ease of distinction and categorisation.

# Gantt Chart

Due to the quick development time of the project, the end date of many tasks may be much further in the future than their actual completion.

The timeline for tasks should therefore be considered a deadline and not a timeline of events.

Furthermore, while the functionality of the system is near completion, it is yet to be tested. As such group members who did not play as large a role in the early development can use this as an opportunity to take weight from the others.

There is also an extra features task, which is intended to be open ended. The purpose is to indicate that upon completion of all essential features and tests, there will likely be room to implement extra functionality. The format this will take is unknown but will be outlined at subsequent group meetings.